School of Engineering Science
Simon Fraser University
hgarg@sfu.ca

March 30, 2017

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser University
Burnaby, BC – V5A 1S6

Re: Design Specifications for NoStress GPS

Dear Dr. Rawicz:

The attached document is the Design Specification for the transparent Heads Up Display (HUD) "NoStress GPS". Our team of NoStress aims to tackle the problem of distracted driving resulting from conventional navigation systems by offering an innovative solution. NoStress GPS is a fully transparent HUD navigation system mounted behind the steering wheel. The transparency of the display allows the driver to clearly see the road ahead, even when using our product.

The purpose of this document is to provide detailed information about the implementation of NoStress GPS. The main part of the design specification document goes over the details of mechanical, hardware and software modules. This document also includes Test Plan and User Interface Appendices. The Test Plan portion reports on the set of test we plan on performing in order to ensure proper functioning of our product. User Interface Design Appendix describes interface of mobile app as well as the GPS module and provides usability analysis.

Our team at NoStress consists of extremely talented and dedicated Engineering Students: Evgeny Kuznetsov, Gur Kohli, Himanshu Garg, and Svetlana Borkovkina. Thank you for your time. For any questions and concerns about the document, please contact us by email at hgarg@sfu.ca.

Sincerely,

Gur Kohli

Gur Kohli
Enclosure: *Design Specifications for NoStress GPS*

Design Specifications for

NoStress GPS

Team Members

Evgeny Kuznetsov
Gur Kohli
Himanshu Garg
Mikael Saad
Svetlana Borkovkina

Contact Person

Himanshu Garg
hgarg@sfu.ca

Submitted To

Steve Whitmore (ENSC 405W)
Dr. Andrew Rawicz (ENSC 440)
School of Engineering Science
Simon Fraser University

Prepared On

March 29, 2017

Abstract

Distracted driving is a leading cause of car crash fatalities in B.C. As per ICBC, on average 81 people die yearly in crashes primarily due to distracted driving. It is responsible for 27% of all car crash fatalities in B.C. [1] NoStress proposes to fix this problem by developing their product, NoStress GPS. Our product will offer a navigation system easier to use than conventional smartphone apps and more convenient and safe than conventional GPS. Our GPS will mount behind the steering wheel and will contain a transparent display that means the driver will never need to take his eyes off the road even for a split second. Connected with and controlled via our smartphone app, the GPS will always stay up-to-date. This document provides an exhaustive list of the functionalities and requirements that are to be followed during the development of NoStress GPS. It includes functional and nonfunctional requirements for proof-of-concept, prototype and the final product stages. It is intended to convey to the stakeholders, an exhaustive list of requirements that shall be followed and standards that shall be maintained during the development of the NoStress GPS.

Table of Contents

Glossary

GPS	Global Positioning System is a space-based radio navigation system that provides geolocation and time information.
LCD	Liquid crystal display, a type of display screen.
HUD	Heads Up Display.
API	Application Program Interface is a set of routines, protocols, and tools for building software applications.
Arduino	Arduino is a computer hardware and software company, project, and user community that designs and manufactures microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical world.
Raspberry Pi	Raspberry Pi is a low-cost, basic computer that was originally intended to help spur interest in computing among school-aged children. The Raspberry Pi is contained on a single circuit board and features ports for: HDMI. USB 2.0. Composite video.
LED	Light-emitting diode, a p-n diode that produces light when current is applied to it.
iOS	An operating system used for mobile devices manufactured by Apple Inc.
Android	An open-source operating system used for smart phones and tablet computers.
Open Source	Denoting software for which the original source code is made freely available and may be redistributed and modified.
Bluetooth	A IEEE 802.15 standard for the short-range wireless interconnection of mobile phones, computers, and other electronic devices.
BLE	Bluetooth Low Energy

Introduction

Scope

This document provides in depth design specifications of "NoStress GPS" transparent display navigation system for general purpose vehicles. The technical details for this document are intended to cover the proof-of-concept stage of the system development including software, hardware and mechanical parts, as well as give essential insights on the future development of the product. Test appendix that is included in this document have necessary information about how each unit, integration, performance and user type of testing that are being held to assure the quality of the product. UI appendix is attached at the end of the document to demonstrate the design of the interface for GPS module and mobile application making main aspect on simplicity and functionality.

Intended Audience

This document is to be used by NoStress engineers. While working on "NoStress GPS" engineers are highly advised to use the document as a primary source of information for implementation and testing of every part of the system. Test Plan appendix is used by engineers to conduct the various testing procedures on a device and make sure it fully follow Requirement Specification document. User interface mockups are provided in order use them later for implementations of the functional UI.

Project Background

"NoStress GPS" navigation system is our proposed solution against distracted driving. People these days rely on some type of navigation system, either using a GPS Navigator or a smartphone for this purpose. However, reading or typing directions on a GPS while using a smartphone application for getting navigation instructions also cause drivers to take their eyes off the road. NoStress GPS is designed to provide a safe and reliable alternative to the current competition and work around their limitations by offering a transparent display coupled with a simple and intuitive user interface, while being affordable to the average user. The primary objective of our product is to allow drivers to navigate while keeping their full attention on the road.

System Specifications

Use case

One of our main goals at NoStress GPS is to provide a device which is easy to use and have a simple user interface. The device shall be mounted on the dashboard right in front of the driver. User then starts the system using the main switch. The user then inputs the destination address in the mobile app and the route is then displayed on the screen with all the necessary information for an intuitive UI.

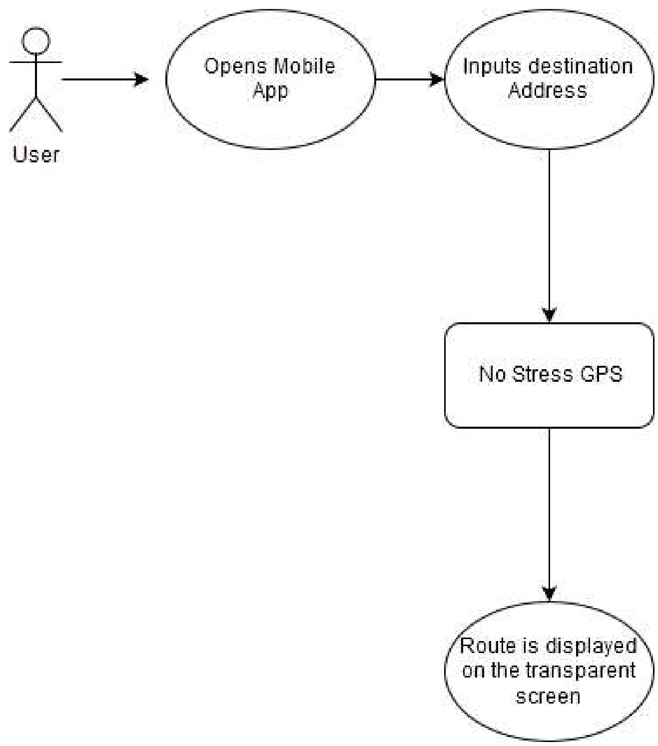

Figure 1: A typical use case

High level design

NoStress GPS consists of a Raspberry Pi board and an Arduino Uno board. The Raspberry Pi is responsible for communicating with the GPS receiver and the Arduino Uno board which controls the screen and other sensors. The phone requests the current coordinates and the user provides with the destination. A request is made to Open Source Routing Machine to calculate the route between the 2 GPS coordinates. The route is then displayed on the screen.

`

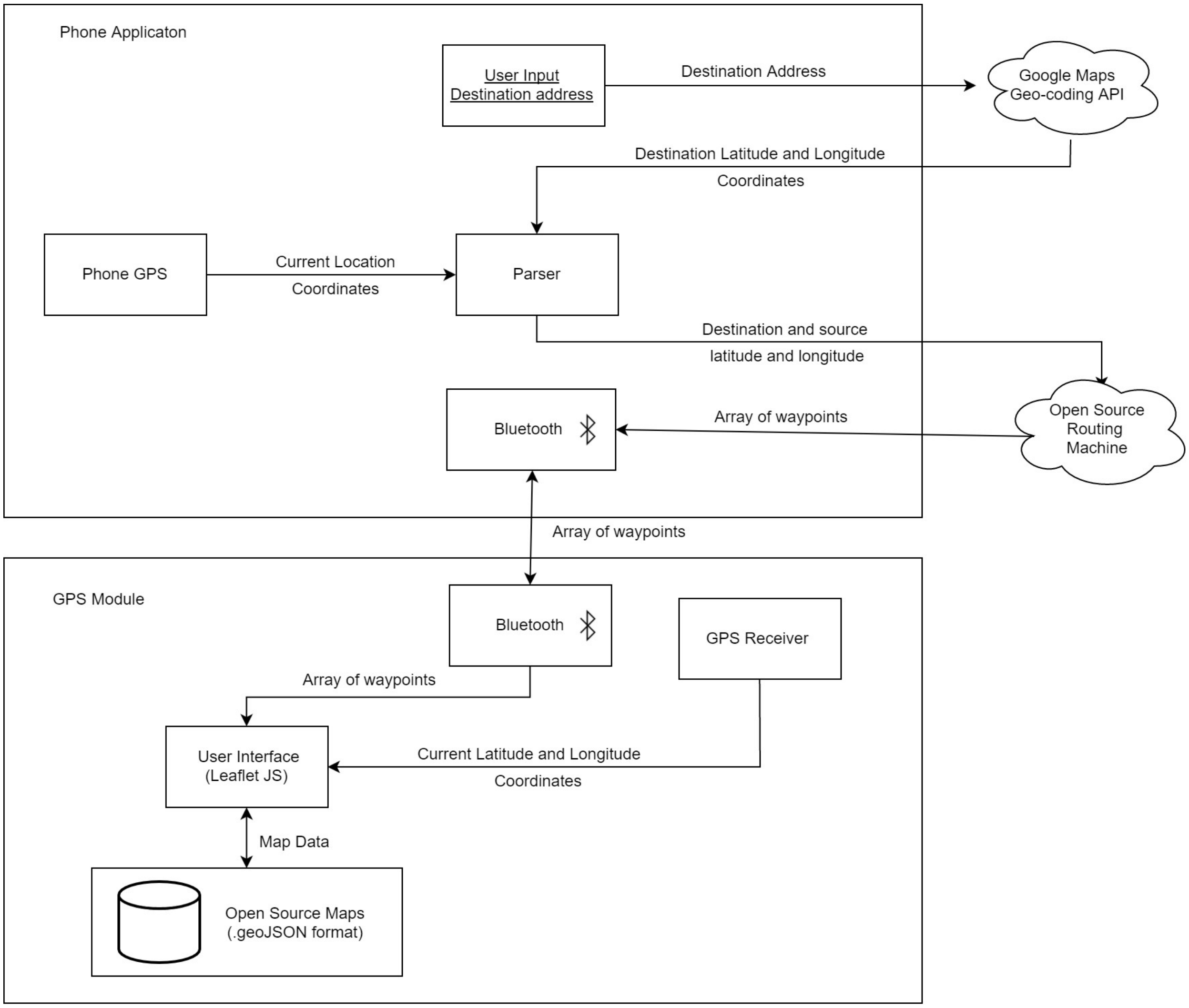

Figure 2: High level design of NoStress GPS

Functional Justification

At NoStress GPS our aim is to provide drivers with a product that is easy to use, less distracting and low cost. Our team decided to make our system portable so that it could be installed in any car/truck dashboard. Our team's primary focus is to develop a minimal UI so that we can provide the driver with just the right amount of information to navigate from one point to another with the least amount of distractions. We also want to provide a cheaper device compared to our competitors.

Our team decided to go with Raspberry Pi as the main controller and Arduino Uno for our secondary controller because of their low cost and open source software available which play a crucial role in reducing the cost of our product. To be sustainable, NoStress GPS uses ABS plastic for its enclosure, recyclable modules and microcontrollers. Using ABS also helps protect the internal components from getting damaged.

Safety and Sustainability

The designers of NoStress GPS are focused on building a safe and sustainable product. The enclosure will be designed with rounded edges to eliminate the danger to the user. All electronic components will be placed in plastic shell to prevent direct contact burning due to heat dissipation.
Our team is also providing the user with a sustainable design which follows cradle to cradle concept. We are committed to providing with a product that has the least negative impact on the environment. The product is designed keeping sustainability in mind at all times

Mechanical Design

Enclosure

The enclosure contains Raspberry Pi, GPS receiver chip, Arduino Uno. The enclosure will have 2 parts, the base and the top cover. The base will contain all the components except the screen. The screen will be embedded on the top of the cover. Dimensions of the transparent screen are: 5.70”x 4.10”x1.50” inches.

Figure 3: Solidworks render of the enclosure case

Hardware Design

LCD Panel

NoStress GPS requires a LCD display bright enough to counter the effects of sunlight. NoStress GPS uses AUO G050VTN01.1 LCD panel that has a typical brightness of 1000 cd/m^2 which is sufficient for bright sunny day conditions. The brightness of the screen is adjusted automatically by sensing the time of day and the ambient light around the GPS module. The LCD panel has a resolution of 800x480 pixels and a diagonal width of 5 inches. The panel will be used in landscape orientation and the contents will be mirrored along the shorter edge so that the reflection is displayed upright on the transparent acrylic sheet.

GPS Receiver

GPS module provides the user with latitude, longitude, current speed, altitude which are used to triangulate the location on the real world. Adafruit Ultimate GPS can track up to 22 satellites on 66 channels with -165dB high sensitivity tracking. [1] It is capable of sampling at 10 times a second and the velocity can be accurate to within 0.1 m/s and position within 3 meters. Since NoStress GPS aims to provide navigation service and real time location updates to our users, these tolerances are within the acceptable values. Adafruit Ultimate GPS module is 25.5mm x 35mm x 6.5mm and weighs 8.5 gram.

Figure 4: GPS Receiver Module [1]

Magnetometer

Magnetometer provides us with the information about where the strongest magnetic forces are coming from often used to detect magnetic north, which is a crucial component of our navigation system. We are using the Adafruit triple-axis accelerometer + magnetometer which will be discussed in detail in the section below.

Accelerometer

Accelerometer is an electromechanical device that measures the acceleration forces. These forces can be static, for instance the force of gravity, or they could be dynamic, for example, if the accelerometer is moved. By measuring the static acceleration action upon it, one can find out what angle is the device tilted at with respect to the ground. [2] We are using the Adafruit Triple-axis Accelerometer + Magnetometer board, which features 3 magnetic field channels and 3 acceleration channels. It can provide from ±1.3 to ±8.1 gauss magnetic field full-scale and ±2g/±4g/±8g/±16g selectable full-scale all selectable by the user. [3]

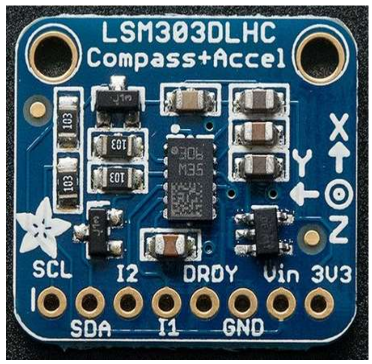

Figure 5: Adafruit Triple-axis Accelerometer + Magnetometer

Bluetooth Chip

In order to transmit data between the GPS module and the app, we are using Bluetooth Low Energy (BLE) technology as it does not consume a lot of power and will not drain phone battery. For proof of concept and prototype we are using Raspberry Pi built-in Bluetooth 4.0 module. For final product, we will be using Adafruit BLE 4.0 nRF8001 Breakout. [3]

Buttons

Figure 6: A tactile button [4]

Five tactile buttons were introduced, in order to give user an ability to interact with GPS module without need of using a mobile application. Each button has ability to sense if user pressing them thus it will close the electrical connection and conduct current. Button have two different states, normally open (NO) when they left idle, and when the button is pressed they are momentarily closed (NC). Buttons are located at front of the GPS module thus making easy access for user even while driving a car. Under normal condition (50mA / 12VDC) buttons can be used up to 300,000 times before breakdown, which is more than enough for the lifetime of the GPS module. [4]

Software Design

Mobile

Mobile application is the main way of communication between the user and the system. The app is available for Android and Apple devices. Through the application, user can specify destination address, change GPS module settings, and update the firmware. We decided to use React Native for the app development. React Native allows us to develop cross-platform application using JavaScript which would significantly reduce development time.

User starts with typing the destination address. The application provides suggestions based on user history. After receiving destination address, it makes a request to a Geocoding API which responds with the latitude and longitudinal coordinates. The application obtains current location from the phone GPS. Current and target coordinates are then forwarded to Open Source Routing Machine which responds with a list of waypoints which give the shortest route from source to destination. These waypoints are sent over to the main controller via Bluetooth. After that, the application is no longer used and the GPS module runs independently of the application.

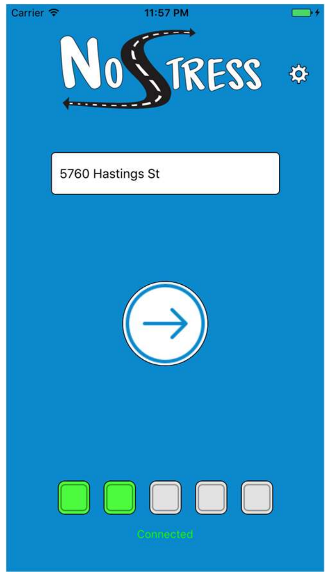

Figure 7: Mockup of mobile app

Bluetooth

We are using React Native BLE manager library to enable connection between the phone and GPS module. [5]

Open Source Routing Machine (OSRM)

OSRM is a routing engine for finding shortest paths in road networks. OSRM calculates the shortest path between origin and destination within a few milliseconds and is free to use.

Maps

NoStress GPS uses the open source "Tileserver-gl-light" JS library developed by Klokan Technologies to serve map data and then Mapbox GL JS library developed by Mapbox to use that data and render tiles on the maps. Mapbox GL JS is then further used to display routes, pan, zoom and is the controller for the primary maps interface.

Firmware

Firmware running on the GPS module has three essential components: handling the communication with user's phone, reading and analyzing data from the sensors and updating the screen with new map frames. Raspberry Pi 3 uses "Bleno" library based on JavaScript interface and written in C++, to perform BLE scanning and data transmission. It allows Raspberry Pi to be visible to other devices in the acceptable Bluetooth range and accepts the connection from user mobile phone. Raspberry Pi is also responsible for rendering and serving map tiles on a screen. In order for maps to be displayed on a screen, Raspberry Pi has to read information from all sensors and make a decision which map tile to render based on given data. After that, frame is sent to the LCD screen with Arduino microcontroller, which is responsible for displaying it on a screen. To provide a good user experience and performance at target frequency of refreshing rate to be 30Hz.

Firmware is implemented with using modular approach which means that code is properly encapsulated and can be run separate threads. There is a main script developed in Python that is responsible to interact with three modules mentioned above trough developed Application Program Interfaces (APIs).

Conclusion

The Design Requirement document clearly lists mechanical, hardware, firmware and software part of GPS module system. Engineers at NoStress are required to follow outline design guidelines as closely as possible. It is expected that sometimes it may not be possible to fully follow the documents and we do expect that as project goes in Beta stage there will be changes in the requirements. For that reason, here at NoStress we practice agile development and design changes is something that we are not anticipating. During weekly team meetings, every member of NoStress provides their feedback on the work that has been done and proposes alternative solution and technologies to be used. In case of any changes, they are approved by the team and documented in the weekly meetings. Our priority is find simplest and fastest solution that will work, and then optimize it for reliability.

As of now, we have finalized the implementation for software and hardware that will be used in final design of the product. We do have an idea of how all mechanical parts will come together and how it will place on a car dashboard above the steering wheel. However, to be more confident, there is a need to conduct some tests to improve the design. The potential design changes may arise after those tests are completed. Our engineers are fully prepared to deal with those issues during the next stage of the product development.

At NoStress, our goal is to take a large portion of the market share by a simple intuitive user interface at an affordable price. With the release of NoStress GPS, we believe we will the market of existing GPS systems and “drive” our product to every household.

References

[1] "Adafruit," 2017. [Online]. Available: https://www.adafruit.com/products/746. [Accessed 2017].

[2] "Dimensionengineering," 2017. [Online]. Available: https://www.dimensionengineering.com/info/accelerometers. [Accessed 2017].

[3] "Adafruit," 2017. [Online]. Available: https://cdn-shop.adafruit.com/datasheets/LSM303DLHC.PDF. [Accessed 2017].

[4] "Arduino," 2017. [Online]. Available: https://www.arduino.cc/documents/datasheets/Button.pdf. [Accessed 2017].

[5] "React-native-ble-manager," 2017. [Online]. Available: https://www.npmjs.com/package/react-native-ble-manager. [Accessed 2017].

[6] "developer.android," 2017. [Online]. Available: https://developer.android.com/guide/practices/ui_guidelines/index.html. [Accessed 2017].

[7] "design-principles," 2017. [Online]. Available: https://developer.apple.com/ios/human-interface-guidelines/overview/design-principles/. [Accessed 2017].

[8] "pcb_tactile_switch_6mm-500x500," 2017. [Online]. Available: http://www.hobbytronics.co.uk/image/cache/data/rapid/pcb_tactile_switch_6mm-500x500.jpg. [Accessed 2017].

[9] "klokantech," 2017. [Online]. Available: https://github.com/klokantech. [Accessed 2017].

[10] "When driving with GPS is against the law," 24 January 2017. [Online]. Available: http://www.consumerreports.org/cro/news/2014/08/when-driving-with-gps-is-against-the-law/index.htm.

[11] ICBC, "Distracted Driving," ICBC, [Online]. Available: http://www.icbc.com/road-safety/crashes-happen/Distracted-driving/Pages/default.aspx. [Accessed 24 January 2017].

[12] A. geeks, "Compare Cars with HUD (head-up display)," [Online]. Available: http://cars.axlegeeks.com/d/f/HUD-%28head-.-up-display%29. [Accessed 29 March 2017].

Test Plan Appendix

Unit test

Unit testing is introduced in order to test every single component of GPS module separately before it is integrated in a design. It includes hardware testing via verification of data communication as well as functionality of the sensors and microcontrollers.

Component	Testing Scenario	Acceptance Criteria
Raspberry Pi microprocessor	Power through USB port from car outlet (5V)	Boot up completed successfully.
Arduino UNO microprocessor	Power through USB port from car outlet (5V)	Boot up completed successfully.
Bluetooth chip	Enable Bluetooth module, pair with mobile phone, write data and then read data	Check that Bluetooth module was found by the phone, and that data was successfully transferred end to end.
GPS Module	Wire up the GPS sensor to Raspberry Pi, read data from sensor.	Raspberry Pi should display current GPS coordinates (Latitude and Longitude) on the console and update it ten times per second.
Accelerometer	Wire up the Accelerometer sensor to Raspberry Pi, read data from sensor.	Raspberry Pi should display X, Y, Z displacement in m/s^2 based on user activity.
Magnetometer	Wire up the Magnetometer sensor to Raspberry Pi, read data from sensor.	Raspberry Pi should display X, Y, Z and heading direction based on user orientation in space.
LCD screen	Connect LCD display to TTY connector to Arduino Uno to Raspberry Pie.	Screen should light up and display the Raspberry Pie console window.
Buttons	Wire up and click a button	Raspberry Pi should register the state of the button at any given moment.

Integration test

The purpose of integration testing is to verify that systems modules function well as a whole. It covers more complex scenarios mainly the ones involved with handling of the data communication from one module to another.

Component	Testing Scenario	Acceptance Criteria
Open route machine API testing	User enters designed coordinates into mobile application and presses send button. Phone makes a request to the. Open route machine API.	Data have been returned to the phone as a set of geographical coordinates and metadata. Phone app is able to parse the data via JSON format so it is ready to be sent to GPS module.
GPS navigation testing	When user presses "Go" button with valid coordinates of a destination.	Phone sends set of geographical coordinates with metadata as a JSON object. GPS module receives data and graphically display route on the map.
Program a button with favorite destination.	User enters one or more favorite destination address in mobile app then clicks configure button.	Data is transmitted from mobile application to GPS module and store in nonvolatile memory of the Raspberry Pi.

Performance testing

Performance testing is required to estimate how system behaves in terms of responsiveness and stability under a particular workload. Performance is an important criterion when evaluating user experience that uses product. It will come into place after prototype stage is completed.

Component	Testing Scenario	Acceptance Criteria
System boot up	Connect the power wire to car electrical outlet. Turn on the car engine to boot up the GPS module	System should boot up and automatically connect to user phone in 10 or less seconds

End to End functionality	Open the mobile application and enter the destination to travel, press go.	After user pressed Go button on the screen, it should take 6 seconds or less for GPS module to display the route on a screen

User acceptance testing

Once a stable version of the product is obtained, it will be required to demonstrate the product to a focus group. It will be achieved by inviting random people to try the product and provide their feedback and potential changes for the product. The goal is to make user acceptance testing early enough so it gives us a chance to reconsider design problems and improve over quality of the product before its release. This type of testing will occur in a real car but under safe environment. There will be a document that will be given for user as an acceptance to participate in testing with brief introduction about scope of the testing. After test drive is complete there will be a document to be completed by user that will provide feedback about user experience and possible improvements. The feedback will be analyzed by engineers and potential changes made for final version of the product.

User Interface Appendix

Introduction

This appendix describes the user interface for NoStress GPS in detail including the testing strategies undertaken and engineering standards followed to create a simple and responsive user interface for both the smartphone application and the GPS module.

Analysis

User Analysis

The following assumptions have been considered when designing the user interface for NoStress GPS

- User owns a smartphone that runs an updated version of Android or iOS operating system.
- User understands how to operate an average smartphone GPS application.
- User understands how an average GPS navigation system for vehicles functions.
- User is of legal driving age and can operate a vehicle.

Technical Analysis

The user interface has been designed following the concept of 7 elements of UI Design proposed by Don Norman in *The Design of Everyday Things.*

Discoverability

Centered on the main page of the smartphone app is the text input field with a descriptive placeholder informing user to enter the destination address. Below the text input field, a button exists to start the trip. A cog icon at the top right of the smartphone app indicates to the user the presence of a settings page.

The smartphone application provides a status of the connection with the GPS module on the front page. It provides a textual as well as a visual indication indicated by a red “Not Connected”, yellow “Connecting” and green “Connected” label.

The smartphone application also provides user with a visual indication of the state of 5 programmable buttons on the GPS module. A button when programmed becomes green to indicate it. Non-programmed buttons are indicated as grey.

The GPS module displays a top down view of the map when a new route is set and the car is stationary. When the sensors sense motion, the GPS shift to a perspective zoomed-in view towards the direction in which the car is travelling. The GPS module displays information about the vehicle's speed, remaining distance to the destination, estimated time of arrival and information about the next route to the user.

Feedback

In the smartphone app, upon entering a new destination in the text field and pressing the "go" button, the user is asked for a confirmation of his actions if there's a previous trip already being displayed on the GPS. The GPS is updated with the new trip.

In the smartphone app, upon entering a new destination in the text field and pressing on one of the programmable buttons, the user is asked for a confirmation to link that button with that destination. The button color updates to indicate action confirmation.

In the smartphone app on the settings page, upon changing the display units, the information on the GPS is updated with the selected units.

Conceptual

The user is offered all the available options within two pages of the smartphone app in a clutter-free format. The user can see the connection status, view, update or enable a programmable button, and start a trip all from the front page. A cog icon on the front page also allows user to open the settings page for further modifications to the smartphone app or the GPS.

Affordances

The user interface for the smartphone application is designed such that user has a clear realization of all the available actions that the user can perform, at the first glance.

Signifiers/ Mappings

Every button or touchable field in the smartphone app provides a visual feedback of being touched by changing its opacity and color when the user's touch is active and reverts to normal color style when no touch is detected. Loading screens are shown by an animated loading icon signifying an action that is being processed in response to user's request.

Constraints

User is only allowed to interact with the GPS module through the smartphone app or the programmable buttons and the reset button on the GPS. The enclosure is sealed constraining the user from accessing the hardware inside the GPS module and ensuring the safety of the user and the product. The smartphone app allows abstraction of the components and services of the GPS module and prevents any unauthorized or unsupported access to the GPS.

Engineering Standards

To ensure easiness of use of the GPS module, we plan on adhering to the following engineering standards when developing user interface:

- ISO 9241 - Ergonomics of Human System Interaction. We will be paying attention to following parts of the standard:
    - ISO 9241-161 Guidance on visual user interface elements
    - ISO 9241-171 Guidance on Software Accessibility
    - ISO 9241-210 Human-centered design for interactive systems
    - ISO 9241-303: Requirements for electronic visual displays
- For app development, we are following Android Design Principles [6] and iOS Human Interface Guidelines. [7]

Usability testing

Analytical

The smartphone application is designed keeping in mind that the user needs to spend the least amount of time on the app, using it only to start a trip on the GPS. For that purpose, the application provides the core actions on the main page of the app in an easy to understand format. The settings page is reserved for actions that are not urgent or actions that do not affect the core functionality of the device. The user is assumed to not be in a hurry when they open the settings page which allows us to put options such as firmware updates there.

The GPS interface provides no configurable options to the user because its primary function is to display maps. This improves the user experience by providing him with functionality that's relevant to them, hiding the unimportant details.

Empirical

Once a stable version of the product is obtained, it will be required to demonstrate the product to a focus group. It will be achieved by inviting random people to try the product and provide their feedback and potential changes for the product. This type of testing will occur in a real car but under safe environment. There will be a document that will be given for user as an acceptance to participate in testing with brief introduction about scope of the testing. After test drive is complete there will be a document to be completed by user that will provide feedback about user experience and possible improvements. The feedback will be analyzed by engineers and potential changes made for final version of the product.

User Interface Mockups

Smartphone application

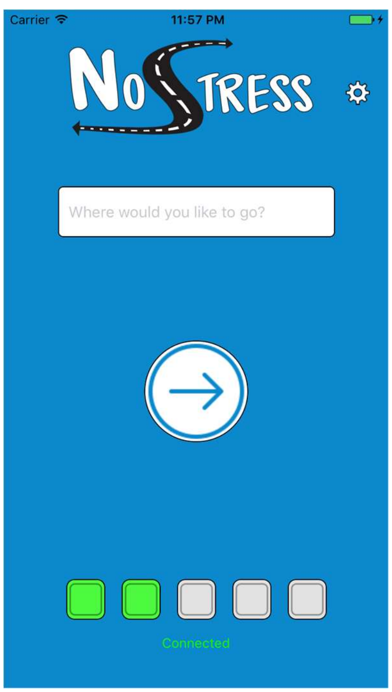

Figure 8: App Home Page

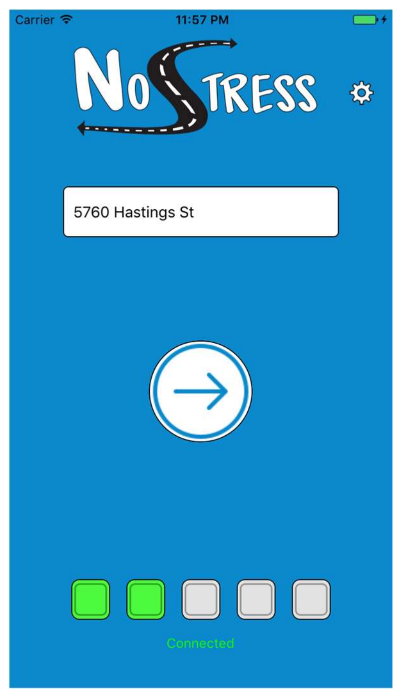

Figure 9: App homepage with address filled in

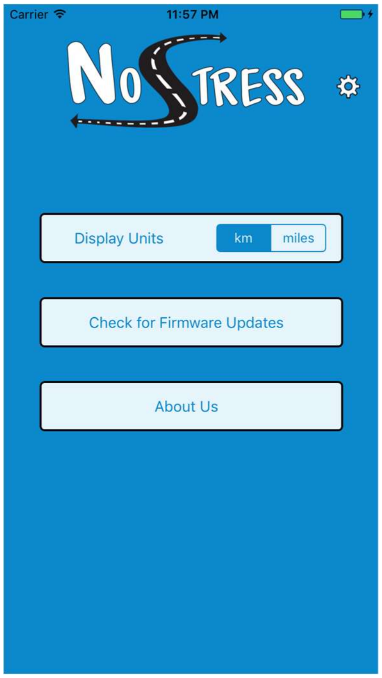

Figure 10: Settings Pane

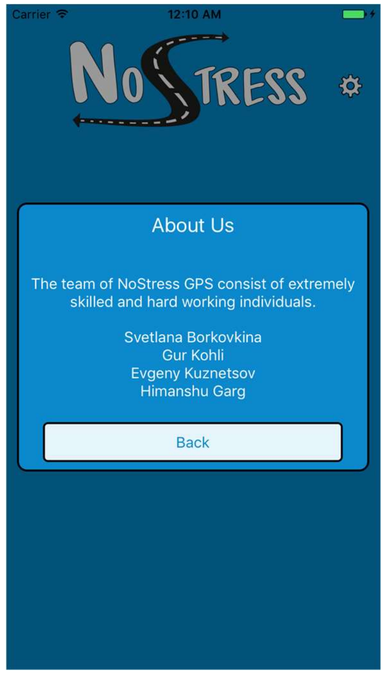

Figure 11: About the team

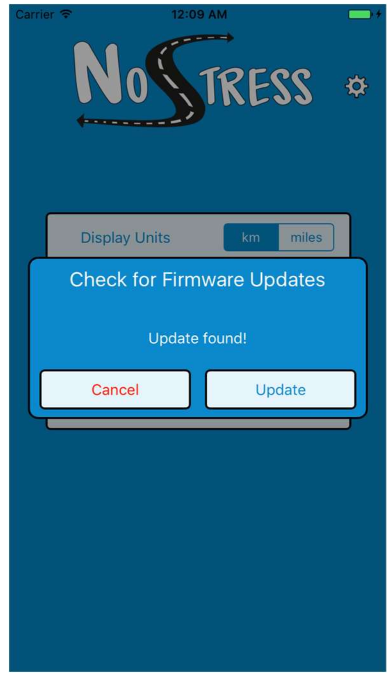

Figure 12: Available Updates

GPS

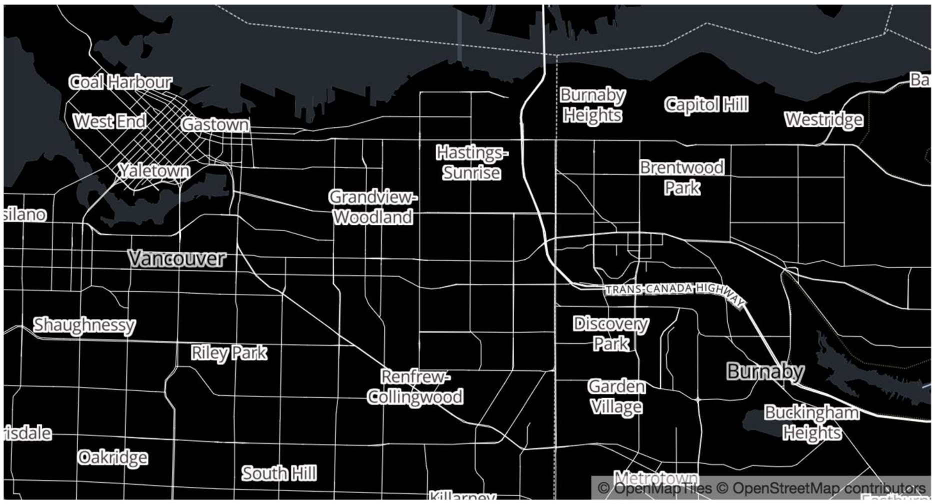

Figure 13: Displayed maps when GPS is powered on for the first.

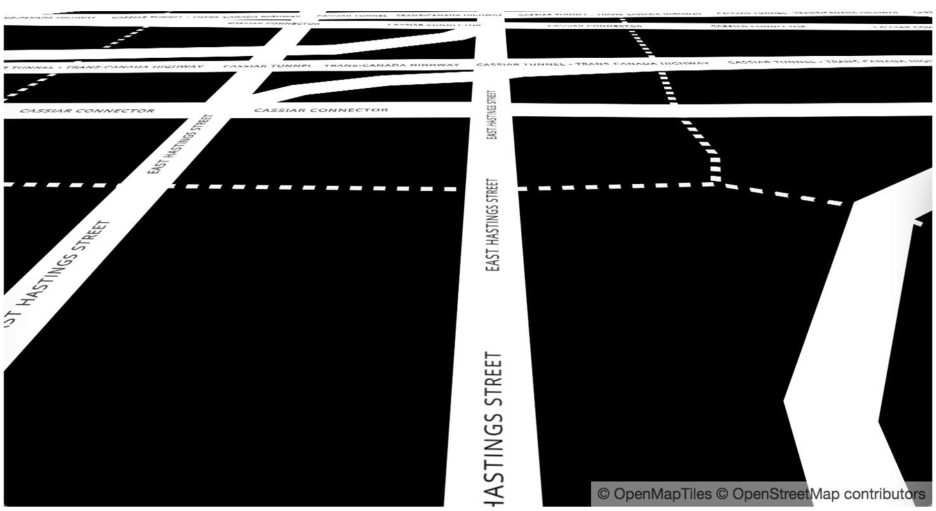

Figure 14:NoStress GPS perspective street view

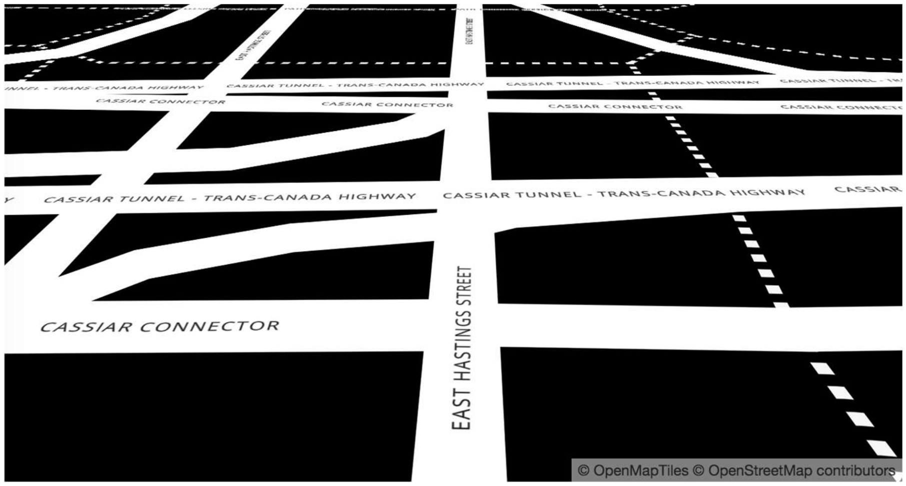

Figure 15: Intersections are now easier to understand with perspective view

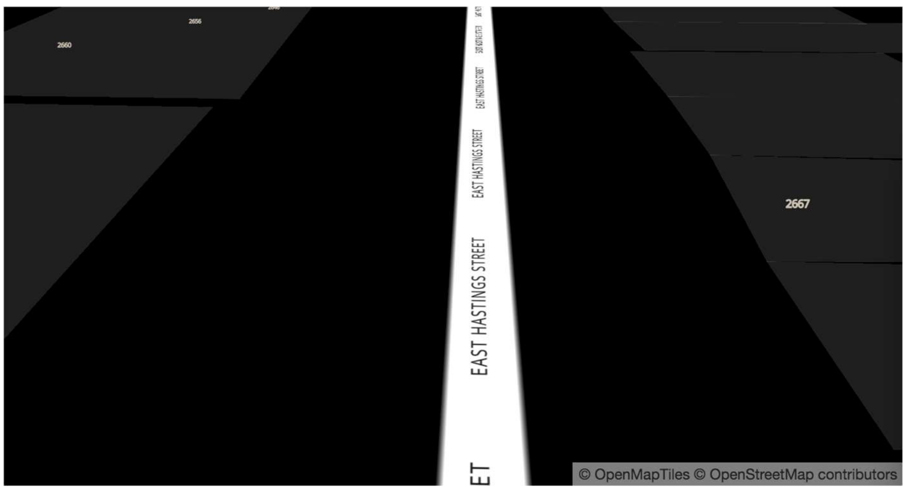

Figure 16: Buildings are displayed with street numbers

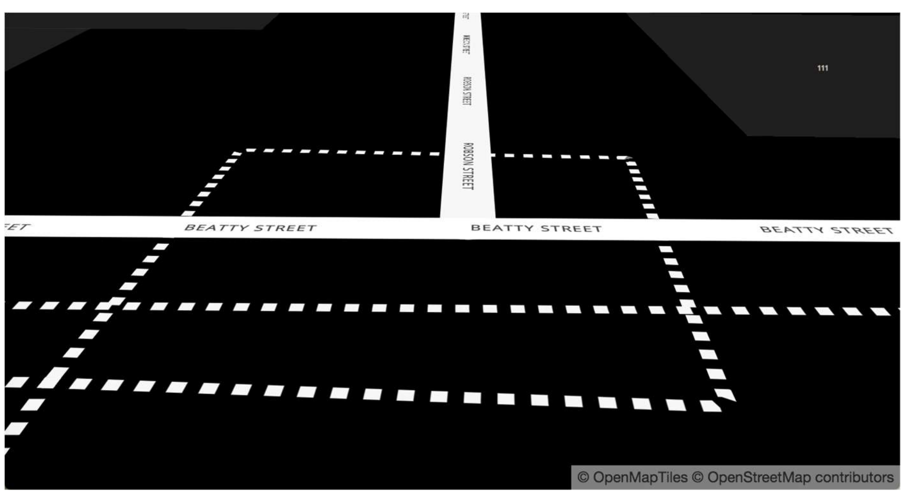

Figure 17:Maps zoom in to perspective view when user is in motion